PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

## Search Results

Search Results for: [look up table and data conversion]
Found 5 of 107,790 searched. → Rerun within the Portal

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score** **Binder**

**Results 1 - 5 of 5 short listing**

**1** Using the co-existence approach to achieve combined functionality of object-oriented and relational systems 80%
R. Ananthanarayanan , V. Gottemukkala , W. Kaefer , T. J. Lehman , H. Pirahesh
ACM SIGMOD Record , Proceedings of the 1993 ACM SIGMOD international conference on Management of data June 1993
Volume 22 Issue 2

Once considered a novelty, object oriented systems have now entered the mainstream. Their impressive performance and rich type systems have created a demand for object oriented features in other areas, such as relational database systems. We believe the current efforts to combine object oriented and relational features into a single hybrid system will fall short of the mark, whereas our approach, the co-existence approach, has the distinction of requiring far less work, but ...

**2** APL2OS: design considerations for a nested array file system 77%
David M. Weintraub
ACM SIGAPL APL Quote Quad , Conference proceedings on APL 90: for the future May 1990
Volume 20 Issue 4

APL2OS is an External Function for the APL2 system, designed to enable APL2 applications to access operating system files (and information about these files) in a straightforward and efficient way, using the power of APL2 syntax to maximum advantage. The design goals and approaches for APL2OS are discussed, in the context of a summary of its features.

**3** Model-driven development of Web applications: the AutoWeb system 77%
Piero Fraternali , Paolo Paolini
ACM Transactions on Information Systems (TOIS) October 2000
Volume 18 Issue 4

This paper describes a methodology for the development of WWW applications and a tool environment specifically tailored for the methodology. The methodology and the development environment are based upon models and techniques already used in the hypermedia, information systems, and software engineering fields, adapted and blended in an original mix. The foundation of the proposal is the conceptual design of WWW applications, using HDM-lite, a notation for the specification of structure, nav ...

**4** VEX: a volume exploratorium: an integrated toolkit for interactive volume visualization 77%
Larry Gelberg , David Kamins , Jeff Vroom
Proceedings of the 1989 Chapel Hill workshop on Volume visualization
May 1989

**5** Object orientation in multidatabase systems 77%
Evaggelia Pitoura , Omran Bukhres , Ahmed Elmagarmid
ACM Computing Surveys (CSUR) June 1995
Volume 27 Issue 2

A multidatabase system (MDBS) is a confederation of preexisting distributed, heterogeneous, and autonomous database systems. There has been a recent proliferation of research suggesting the application of object-oriented techniques to facilitate the complex task of designing and implementing MDBSs. Although this approach seems promising, the lack of a general framework impedes any further development. The goal of this paper is to provide a concrete analysis and categorization of the various ...

**Results 1 - 5 of 5** **short listing**

PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

# Search Results

Search Results for: [interpolation and data conversion]
Found 29 of 107,790 searched. → Rerun within the Portal

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score Binder**

**Results 1 - 20 of 29 short listing**

Prev Page 1 2 Next Page

1 Remark on algorithm 123 80%
Stephen P. Barton , John F. Wagner
Communications of the ACM March 1964
Volume 7 Issue 3

2 Certification of algorithm 122: Tridiagonal matrix 80%
Peter Naur
Communications of the ACM March 1964
Volume 7 Issue 3

3 Algorithm 222: Incomplete beta functions ratios 80%
Walter Gautschi
Communications of the ACM March 1964
Volume 7 Issue 3

4 Algorithm 221: Gamma functions 80%
Walter Gautschi
Communications of the ACM March 1964
Volume 7 Issue 3

5 A cost-benefit decision model: analysis, comparison amd selection of data management 80%

Stanley Y. W. Su , Jozo Dujmovic , D. S. Batory , S. B. Navathe , Richard Elnicki
ACM Transactions on Database Systems (TODS) September 1987
Volume 12 Issue 3

This paper describes a general cost-benefit decision model that is applicable to the evaluation, comparison, and selection of alternative products with a multiplicity of features, such as complex computer systems. The application of this model is explained and illustrated using the selection of data management systems as an example. The model has the following features: (1) it is mathematically based on an extended continuous logic and a theory of complex criteria; (2) the decisi ...

**6** Representation of contours and regions for efficient computer search 80%
R. D. Merrill
Communications of the ACM February 1973
Volume 16 Issue 2

A novel computer-searchable representation for the three basic pictorial features, contour maps, region coverage, and line structures, is described. The representation, which has practical storage requirements, provides a rapid means of searching large files for data associated with geometric position as well as with attribute value. An application of this representation to handling terrain information illustrates its utility. The algebraic properties of the data structure make it computati ...

**7** SmartFiles: an OO approach to data file interoperability 80%
Matthew Haines , Piyush Mehrotra , John Van Rosendale
ACM SIGPLAN Notices , Proceedings of the tenth annual conference on Object-oriented programming systems, languages, and applications October 1995
Volume 30 Issue 10

Data files for scientific and engineering codes typically consist of a series of raw data values whose description is buried in the programs that interact with these files. In this situation, making even minor changes in the file structure or sharing files between programs (interoperability) can only be done after careful examination of the data files and the I/O statements of the programs interacting with this file. In short, scientific data files lack self-description, and other self-describin ...

**8** The i860TM 64-bit supercomputing microprocessor 77%
L. Kohn , N. Margulis
Proceedings of the 1989 ACM/IEEE conference on Supercomputing

August 1989

The Intel i860TM processor is a RISC-based microprocessor incorporating a RISC core with memory management, a floating point unit, and caches on a single chip. The 1,000,000 transistors allow a single chip implementation with highly optimized interunit communication and wide internal data buses. The parallelism and pipelining between the execution units, and the innovative cache management techniques are under explicit control of software. Vectorizable applications can ...

**9** Performance evaluation: Concepts of a data base simulation language 77%

Peter Scheuermann

Proceedings of the 1977 ACM SIGMOD international conference on Management of data August 1977

Performance modelling of data base systems requires taking into consideration the complex interactions between the different physical design parameters and the system workload parameters. In order to facilitate a data base designer in evaluating various implementation strategies, a simulation language is presented which has three distinct components (1) data definition (2) query definition and (3) mapping to storage definition. A number of features characterize this type of descriptive mechanism ...

**10** Computer simulation of high-speed impact response of composites 77%

Scott Langlie , Wing Cheng , Ilhan Dilber

Proceedings of the 1990 ACM/IEEE conference on Supercomputing November 1990

This paper shares some of our experiences in the use of a supercomputer to facilitate the analysis and design of high velocity impact process in composite systems, and also in the benefits of color animation of the results on a MacIntosh computer. The calculations were carried out on a CRAY-2 using an impact model developed for high velocity impact/penetration of fiber-reinforced layered composites. The model is built on a continuum approach and includes a basic orthotropic constitutive material ...

**11** Virtual extension: Java distributed objects for numerical visualization in VisAD 77%

William Hibbard , Curtis Rueden , Steve Emmerson , Tom Rink , David Glowacki , Tom Whittaker , Don Murray , David Fulker , John Anderson

April 2002

The scientific world is evolving to require more collaboration among different institutions and disciplines. Understanding long-term changes in the Earth environment, for example, requires models that integrate disciplines such as meteorology, oceanography, hydrology (rivers and groundwater), soil science and geology. During the past 15 years, scientists have started sharing data using FTP and software on the Internet, but collaborative work and more routine data sharing require a new kind of sc ...

**12** Kizamu: a system for sculpting digital characters 77%
Ronald N. Perry , Sarah F. Frisken
Proceedings of the 28th annual conference on Computer graphics and interactive techniques August 2001

This paper presents Kizamu, a computer-based sculpting system for creating digital characters for the entertainment industry. Kizamu incorporates a blend of new algorithms, significant technical advances, and novel user interaction paradigms into a system that is both powerful and unique.

To meet the demands of high-end digital character design, Kizamu addresses three requirements posed to us by a major production studio. First, animators and artists want *digital clay* &mdash; a ...

**13** Scientific visualization of water quality in the Chesapeake Bay 77%
Robert Stein , Alan M. Shih , M. Pauline Baker , Carl F. Cerco , Mark R. Noel
Proceedings of the conference on Visualization '00 October 2000

**14** Remarks on algorithm 32: multint: certification of algorithm 32 77%
K. S. Kölbig
Communications of the ACM December 1968
Volume 11 Issue 12

**15** Algorithm 343: eigenvalues and eigenvectors of a real general matrix 77%
J. Grad , M. A. Brebner
Communications of the ACM December 1968
Volume 11 Issue 12

**16** Algorithms: Algorithm 342: generator of random numbers satisfying the Poisson distribution 77%
Richard H. Show
Communications of the ACM December 1968
Volume 11 Issue 12

**17** Algorithms: Algorithm 325:Adjustment of the inverse of a symmetric matrix when two symmetric elements are changed 77%
Gerhard Zielke
Communications of the ACM February 1968
Volume 11 Issue 2

**18** Algorithms: Algorithm 324: Maxflow 77%
G. Bayer
Communications of the ACM February 1968
Volume 11 Issue 2

**19** Algorithms: Algorithm 323:Generation of permutations in lexicographic order 77%
R. J. Ord-Smith
Communications of the ACM February 1968
Volume 11 Issue 2

**20** Algorithms: Algorithm 322:F-distribution 77%
Egon Dorrer
Communications of the ACM February 1968
Volume 11 Issue 2

**Results 1 - 20 of 29** **short listing**

Prev Page 1 2 Next Page

PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

# Search Results

Search Results for: [interpolation and data conversion]
Found 29 of 107,790 searched. →Rerun within the Portal

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score** Binder

**Results 21 - 29 of 29 short listing**

Prev Page 1 2 Next Page

**21** Algorithms: Algorithm 321:t-test probabilities 77%
John Morris
Communications of the ACM February 1968
Volume 11 Issue 2

**22** Algorithms: Algorithm 320:Harmonic analysis for symmetrically distributed data 77%
D. B. Hunter
Communications of the ACM February 1968
Volume 11 Issue 2

**23** A spatial hierarchical compression method for 3D streaming animation 77%
Toshiki Hijiri , Kazuhiro Nishitani , Tim Cornish , Toshiya Naka , Shigeo Asahara
Proceedings of the fifth symposium on Virtual reality modeling language (Web3D-VRML) February 2000

When distributing 3D contents real-time over a network with a narrow bandwidth such as a telephone line, methods for streaming and data compression can be considered indispensable.In previous work, we made possible the real-time streaming of 3D animation data on a network with a narrow bandwidth such as a telephone line by partitioning motion data for humanoid characters (data obtained by motion capture, for example full frame data at 30 frames/sec) into packets and then carrying ...

**24** VEX: a volume exploratorium: an integrated toolkit for interactive volume visualization 77%
Larry Gelberg , David Kamins , Jeff Vroom
Proceedings of the 1989 Chapel Hill workshop on Volume visualization May 1989

**25** Multiple representations in GIS: materialization through map generalization, geometric, and spatial analysis operations 77%
Clodoveu A. Davis , Alberto H. F. Laender
Proceedings of the seventh ACM international symposium on Advances in geographic information systems November 1999

**26** Describing free-form 3D surfaces for animation 77%
Eben Ostby
Proceedings of the 1986 workshop on Interactive 3D graphics January 1987

A system for interactively describing and modifying free-form surfaces is presented. The system is based on the use of bicubic patches. Although it is not a full-fledged mechanical CAD system, it has been used to construct complex surface descriptions. It is also useful as a testbed for further experimentation.

**27** Supporting heterogeneous data import for data visualization 77%
R. Ford , R. Thompson , D. Thompson
Proceedings of the 1995 ACM symposium on Applied computing February 1995

**28** Using semantic values to facilitate interoperability among heterogeneous information systems 77%
Edward Sciore , Michael Siegel , Arnon Rosenthal
ACM Transactions on Database Systems (TODS) June 1994 Volume 19 Issue 2

Large organizations need to exchange information among many separately developed systems. In order for this exchange to be useful, the individual systems must agree on the meaning of their exchanged data. That is, the organization must ensure semantic interoperability. This paper provides a theory of semantic values as a unit of exchange that facilitates semantic interoperability betweeen heterogeneous information systems. We show how semantic values can ei ...

**29** Reality Engine graphics 77%
Kurt Akeley
Proceedings of the 20th annual conference on Computer graphics and interactive techniques September 1993

**Results 21 - 29 of 29** **short listing**

Prev Page **1** **2** Next Page

PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

## Search Results

Search Results for: [floating point and conversion]
Found 1,435 of 107,790 searched. → Rerun within the Portal

**Warning: Maximum result set of 200 exceeded. Consider refining.**

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score**

**Results 1 - 20 of 200 short listing**

Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page

**1** A precision- and range-independent tool for testing floating-point arithmetic II: conversions 100%
Brigitte Verdonk , Annie Cuyt , Dennis Verschaeren
ACM Transactions on Mathematical Software (TOMS) March 2001
Volume 27 Issue 1

The IEEE 754 and 854 standards for floating-point arithmetic are essentially a specification of a programming environment, encompassing aspects from computer hardware, operating systems, and compilers to programming languages (see especially Section 8). Parts I and II of this paper together describe a tool to test floating-point implementations of arbitrary precision and exponent range (hardware as well as software) for compliance with the principles outlined in the IEEE standards. The tool ...

**2** On the need for predictable floating-point arithmetic in the programming languages Fortran 90 and C/C++ 99%
Dennis Verschaeren , Annie Cuyt , Brigitte Verdonk
ACM SIGPLAN Notices March 1997
Volume 32 Issue 3

During the past decade the IEEE 754 standard for binary floating-point arithmetic has been very successful. Many of today's hardware platforms conform to this standard and compilers should make floating-point functionality available to programmers through high-level programming languages. They should also implement certain features of the IEEE 754 standard in software, such as the input and output conversions to and from the internal binary representation of numbers. In this report, a number of ...

**3** How to print floating-point numbers accurately 99%
Guy L. Steele , Jon L. White
ACM SIGPLAN Notices , Proceedings of the conference on Programming language design and implementation June 1990
Volume 25 Issue 6

We present algorithms for accurately converting floating-point numbers to decimal representation. The key idea is to carry along with the computation an explicit representation of the required rounding accuracy. We begin with the simpler problem of converting fixed-point fractions. A modification of the well-known algorithm for radix-conversion of fixed-point fractions by multiplication explicitly determines when to terminate the conversion process; a variable number of digits ar ...

**4** A Portable Extended Precision Arithmetic Package and Library with Fortran Precompiler 97%
W. T. Wyatt , D. W. Lozier , D. J. Orser
ACM Transactions on Mathematical Software (TOMS) September 1976
Volume 2 Issue 3

**5** A precision- and range-independent tool for testing floating-point arithmetic I: basic operations, square root, and remainder 96%
Bridgitte Verdonk , Annie Cuyt , Dennis Verschaeren
ACM Transactions on Mathematical Software (TOMS) March 2001
Volume 27 Issue 1

This paper introduces a precision- and range-independent tool for testing the compliance of hardware or software implementations of (multiprecision) floating-point arithmetic with the principles of the IEEE standards 754 and 854. The tool consists of a driver program, offering many options to test only specific aspects of the IEEE standards, and a large set of test vectors, encoded in a precision-independent syntax to allow the testing of basic and extended hardware formats as well as multi ...

**6** Integrating SIMD into the undergraduate curriculum 96%
W. D. Maurer
The Journal of Computing in Small Colleges , Proceedings of the sixth annual CCSC northeastern conference on The journal of computing in small colleges April 2001
Volume 16 Issue 4

Assembly language instruction today, in our view, should include instruction in the newly important area of single-instruction, multiple-data (SIMD) instructions. Such instructions are available on all major platforms, and they considerably speed up operations on arrays, particularly large arrays. This speedup is more pronounced with assembly language than with algebraic language programming, and thus provides another reason for undergraduate students to learn assembly language. We discuss th ...

**7** What every computer scientist should know about floating-point arithmetic 95%
David Goldberg
ACM Computing Surveys (CSUR) March 1991
Volume 23 Issue 1

Floating-point arithmetic is considered as esoteric subject by many people. This is rather surprising, because floating-point is ubiquitous in computer systems: Almost every language has a floating-point datatype; computers from PCs to supercomputers have floating-point accelerators; most compilers will be called upon to compile floating-point algorithms from time to time; and virtually every operating system must respond to floating-point exceptions such as overflow. This paper presents a ...

**8** Leo: a system for cost effective 3D shaded graphics 95%
Michael F. Deering , Scott R. Nelson
Proceedings of the 20th annual conference on Computer graphics and interactive techniques September 1993

**9** An optimizing compiler for lexically scoped LISP 95%
Rodney A. Brooks , Richard P. Gabriel , Guy L. Steele
Proceedings of the SIGPLAN '82 symposium on Compiler construction June 1982

We are developing an optimizing compiler for a dialect of the LISP language. The current target architecture is the S-I, a multiprocessing supercomputer designed at Lawrence Livermore National Laboratory. While LISP is usually thought of as a language primarily for symbolic processing and list manipulation, this compiler is also intended to compete with the S-1 PASCAL and FORTRAN compilers for quality of compiled numerical code. The S-1 is designed for extremely high-speed signal processing ...

**10** The design of floating-point data types 94%
David Goldberg
ACM Letters on Programming Languages and Systems (LOPLAS) June 1992
Volume 1 Issue 2

The issues involved in designing the floating-point part of a programming language are discussed. Looking at the language specifications for most existing languages might suggest that this design involves only trivial issues, such as whether to have one or two types of REALs or how to name the functions that convert from INTEGER to REAL. It is shown that there are more significant semantic issues involved. After discussing the trade-offs for the major design decisions, they are illustrated ...

**11** Multiple precision floating-point conversion from decimal-to-binary and vice versa 94%
O. G. Mancino
Communications of the ACM May 1966
Volume 9 Issue 5

**12** Guidance for the use of the Ada programming language in high integrity systems 93%
B. A. Wichmann
ACM SIGAda Ada Letters July 1998
Volume XVIII Issue 4

This paper is the current result of a study by the ISO HRG Rapporteur group which is being circulated for comment. Many people have contributed to this, but those who have either attended two recent meetings of group or have made substantial e-mail comments are: Praful V Bhansali (Boeing, USA), Alan Burns (University of York, UK), Bernard Carre' (Praxis Critical Systems, UK), Dan Craigen (ORA, Canada), Nick Johnson MoD, UK), Stephen Michell (Canada), Gilles Motet (DGEI/INSA, France), George Roma ...

**13** Optimization for a superscalar out-of-order machine 93%
Anne M. Holler
Proceedings of the 29th annual ACM/IEEE international symposium on Microarchitecture December 1996

**14** In-and-out conversions 92%
David W. Matula
Communications of the ACM January 1968
Volume 11 Issue 1

**15** A type-based compiler for standard ML 92%
Zhong Shao , Andrew W. Appel
ACM SIGPLAN Notices , Proceedings of the conference on Programming language design and implementation June 1995
Volume 30 Issue 6

Compile-time type information should be valuable in efficient compilation of statically typed functional languages such as Standard ML. But how should type-directed compilation work in real compilers, and how much performance gain will type-based optimizations yield? In order to support more efficient data representations and gain more experience about type-directed compilation, we have implemented a new type-based middle end and back end for the Standard ML of New Jersey compiler. We descr ...

**16** A set of Ada packages for high-precision calculations 92%
B. G. S. Doman , C. J. Pursglove , W. M. Coen
ACM Transactions on Mathematical Software (TOMS) December 1995
Volume 21 Issue 4

The packages described here are designed to perform efficient high-accuracy calculations in cases where there can be serious loss of significance due to rounding errors. Numbers are represented by a value part with a variable number of digits in the mantissa and an error estimate which is updated throughout the calculation and which gives a range of possible values for the result of the calculation. For economy and speed, intermediate results are truncated so that the least-significant digi ...

**17** On conventions for systems of numerical representation 92%
Peter M. Neely
Proceedings of the ACM annual conference - Volume 2 August 1972

Present conventions for numeric representation are considered inadequate to serve the needs of applied computing. Thus an augmented digital number system is proposed for use in programming languages and in digital computers. Special symbols are proposed for numbers too large, too small or too close to zero to be represented in the normal digital number system, or which are undefined. Properties of mappings among and between digital number systems are used to justify the augments chosen. Fin ...

**18** High-performance polygon rendering 92%
Kurt Akeley , Tom Jermoluk
ACM SIGGRAPH Computer Graphics , Proceedings of the 15th annual conference on Computer graphics and interactive techniques June 1988
Volume 22 Issue 4

**19** Compiling C for vectorization, parallelization, and inline expansion 91%
R. Allen , S. Johnson
Proceedings of the SIGPLAN'88 conference on Programming Language design and Implementation June 1988

Practical implementations of real languages are often an excellent way of testing the applicability of theoretical principles. Many stresses and strains arise from fitting practicalities, such as performance and standard compatibility, to theoretical models and methods. These stresses and strains are valuable sources of new research and insight, as well as an oft-needed check on the egos of theoreticians. Two fertile areas that are often explored by implementations are

**20** System architectures for computer music 91%
John W. Gordon
ACM Computing Surveys (CSUR) June 1985
Volume 17 Issue 2

Computer music is a relatively new field. While a large proportion of the public is aware of computer music in one form or another, there seems to be a need for a better understanding of its capabilities and limitations in terms of synthesis, performance, and recording hardware. This article addresses that need by surveying and discussing the architecture of existing computer music systems. System requirements vary according to what the system will be used for. Common uses for co ...

**Results 1 - 20 of 200** **short listing**

Prev Page **1 2 3 4 5 6 7 8 9 10** Next Page

PORTAL
THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

# Search Results

Search Results for: [floating point and conversion]
Found 1,435 of 107,790 searched. → Rerun within the Portal

**Warning: Maximum result set of 200 exceeded. Consider refining.**

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score**

**Results 21 - 40 of 200 short listing**

Prev Page **1 2 3 4 5 6 7 8 9 10** Next Page

**21** A C language extension for machine-independent programming 91%
Shingo Kamiya , Toshiyuki Yoshida , Takanobu Sugiyasu , Koki Miyazawa
Proceedings of the 1986 ACM SIGSMALL/PC symposium on Small systems December 1986

MIC (Machine-Independent C) is an extension of the C language which has been designed to write portable programs as installed in various small computers. MIC provides unified semantics suitable for typical small computers with new facilities for machine-independent data definition, and its syntax conforms to the preliminary draft of the proposed ANSI standard for C. It is fully implemented as a compiler front end called MICP, and has been applied to actual programming. The principal feature ...

**22** Area and performance tradeoffs in floating-point divide and square-root implementations 91%
Peter Soderquist , Miriam Leeser
ACM Computing Surveys (CSUR) September 1996
Volume 28 Issue 3

Floating-point divide and square-root operations are essential to many scientific and engineering applications, and are required in all computer systems that support the IEEE floating-point standard. Yet many current microprocessors provide only weak support for these operations. The latency and throughput of division are typically far inferior to those of floating-point addition and multiplication, and square-root performance is often even lower. This article argues the case for high-perf ...

**23** A display system for the Stellar graphics supercomputer model GS1000 90%
Brian Apgar , Bret Bersack , Abraham Mammen
ACM SIGGRAPH Computer Graphics , Proceedings of the 15th annual conference on Computer graphics and interactive techniques June 1988
Volume 22 Issue 4

**24** Algorithm 719; Multiprecision translation and execution of FORTRAN programs 90%
David H. Bailey
ACM Transactions on Mathematical Software (TOMS) September 1993

Volume 19 Issue 3

This paper describes two Fortran utilities for multiprecision computation. The first is a package of Fortran subroutines that perform a variety of arithmetic operations and transcendental functions on floating point numbers of arbitrarily high precision. This package is in some cases over 200 times faster than that of certain other packages that have been developed for this purpose. The second utility is a translator program, which facilitates the conversion of ordinary Fortran p ...

**25** Efficient algorithms for 3D scan-conversion of parametric curves, surfaces, and volumes 89%
Arie Kaufman
ACM SIGGRAPH Computer Graphics , Proceedings of the 14th annual conference on Computer graphics and interactive techniques August 1987
Volume 21 Issue 4

**26** An implementation of multiple modulus arithmetic and its comparison to floating point arithmetic 89%
T. M. Rao , Lawrence Helber
Proceedings of the 1993 ACM conference on Computer science March 1993

Multiple-modulus arithmetic, which allows error-free computation for rational inputs, is implemented in a software package called ZIP. Experimentation is carried out comparing floating point and multi-mod arithmetic.

**27** Storage use analysis and its applications 89%
Manuel Serrano , Marc Feeley
ACM SIGPLAN Notices , Proceedings of the first ACM SIGPLAN international conference on Functional programming June 1996
Volume 31 Issue 6

In this paper we present a new program analysis method which we call *Storage Use Analysis*. This analysis deduces how objects are used by the program and allows the optimization of their allocation. This analysis can be applied to both statically typed languages (e.g. ML) and latently typed languages (e.g. Scheme). It handles side-effects, higher order functions, separate compilation and does not require CPS transformation. We show the application of our analysis to two important optimizat ...

**28** VAX floating point: a solid foundation for numerical computation 89%
Mary Payne , Dileep Bhandarkar
ACM SIGARCH Computer Architecture News June 1980
Volume 8 Issue 4

**29** Arithmetic: A flexible floating-point format for optimizing data-paths and operators in FPGA based DSPs 88%
J. Dido , N. Geraudie , L. Loiseau , O. Payeur , Y. Savaria , D. Poirier
Tenth ACM International Symposium on Field-Programmable Gate Arrays February 2002

Video signal processing requires complex algorithms performing many basic operations on a video stream. To perform these calculations in real-time in a FPGA, we must use innovative structures to meet speed requirements while managing complexity. As part of a project aiming at the development of a video noise reducer, we developed an optimized processing stream that required some floating-point calculations. This paper presents the rationale for developing a floating-point unit, justifies the dat ...

**30** The Ada issues: A readers' guide to the Ada issues 88%
Erhard Ploedereder
ACM SIGAda Ada Letters May 1998
Volume XVIII Issue 3

**31** On a storage mapping function for data structures 88%

Phillip Deuel
Communications of the ACM May 1966
Volume 9 Issue 5

**32** TIL: a type-directed optimizing compiler for ML 88%
D. Tarditi , G. Morrisett , P. Cheng , C. Stone , R. Harper , P. Lee
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN '96 conference on Programming language design and implementation May 1996
Volume 31 Issue 5

**33** Comparison of the functional power of APL2 and FORTRAN 90 88%
Robert G. Willhoft
ACM SIGAPL APL Quote Quad , Proceedings of the international conference on APL '91 July 1991
Volume 21 Issue 4

APL and Fortran, although very different, share the challenge of remaining "competitive" in the light of an onslaught of "modern" computer languages. To meet this challenge, both have attempted to enhance their position by adding significant new features to their language. For example, APL2 is an extension of APL.Fortran has also attempted to meet the challenges of modern programming by developing a new Fortran standard called Fortran 90. This standard revises many areas of Fortran, but this pap ...

**34** Representational error in binary and decimal numbering systems 88%
Paul Johnstone
Proceedings of the 20th annual Southeast regional conference April 1982

**35** Information systems development in Ada 88%
B. Brosgol , R. Eachus , D. Emery
Proceedings of the eleventh annual Washington Ada symposium & summer ACM SIGAda meeting on Ada July 1994

**36** Resource allocation in a high clock rate microprocessor 88%
Michael Upton , Thomas Huff , Trevor Mudge , Richard Brown
ACM SIGPLAN Notices , Proceedings of the sixth international conference on Architectural support for programming languages and operating systems November 1994
Volume 29 Issue 11

This paper discusses the design of a high clock rate (300MHz) processor. The architecture is described, and the goals for the design are explained. The performance of three processor models is evaluated using trace-driven simulation. A cost model is used to estimate the resources required to build processors with varying sizes of on-chip memories, in both single and dual issue models. Recommendations are then made to increase the effectiveness of each of the models.

**37** The MAD definition facility 88%
Bruce W. Arden , Bernard A. Galler , Robert M. Graham
Communications of the ACM August 1969
Volume 12 Issue 8

One of the first definition facilities for higher level languages is described. Users of the language can define new operators and/or data types into the MAD language, so that their use appears as if they were predefined. Information is given on how one writes definitions, as well as on much of the motivation behind the form in which definitions are written. Some conclusions are drawn about future definitional facilities.

**38** Compiler transformations for high-performance computing 88%
David F. Bacon , Susan L. Graham , Oliver J. Sharp
ACM Computing Surveys (CSUR) December 1994
Volume 26 Issue 4

In the last three decades a large number of compiler transformations for optimizing programs have been implemented. Most optimizations for uniprocessors reduce the number of instructions executed by the program using transformations based on the analysis of scalar quantities and data-flow techniques. In contrast, optimizations for high-performance superscalar, vector, and parallel processors maximize parallelism and memory locality with transformations that rely on tracking the properties o ...

**39** Applications: A voxel-based parallel collision detection algorithm 87%
Orion Sky Lawlor , Laxmikant V. Kalée
Proceedings of the 16th international conference on Supercomputing
June 2002

Two physical objects cannot occupy the same space at the same time. Simulated physical objects do not naturally obey this constraint. Instead, we must detect when two objects have collided---we must perform collision detection. This work presents a simple voxel-based collision detection algorithm, an efficient parallel implementation of the algorithm, and performance results.

**40** Kava: a Java dialect with a uniform object model for lightweight classes 87%
David F. Bacon

Proceedings of the 2001 joint ACM-ISCOPE conference on Java Grande June 2001

Object-oriented programming languages have always distinguished between &ldquo;primitive&rdquo; and &ldquo;user-defined&rdquo; data types, and in the case of languages like C++ and Java, the primitives are not even treated as objects, further fragmenting the programming model. The distinction is especially problematic when a particular programming community requires primitive-level support for a new data type, as for complex, intervals, fixed-pointed numbers, and so on.

We present Kav ...

**Results 21 - 40 of 200** **short listing**

Prev Page **1 2 3 4 5 6 7 8 9 10** Next Page

PORTAL THE ACM DIGITAL LIBRARY

> home > about > feedback > login
US Patent & Trademark Office

## Search Results

Search Results for: [floating point and conversion]
Found 1,435 of 107,790 searched. → Rerun within the Portal

**Warning: Maximum result set of 200 exceeded. Consider refining.**

Search within Results

> Advanced Search > Search Help/Tips

**Sort by: Title Publication Publication Date Score**

**Results 181 - 200 of 200 short listing**

Prev Page 1 2 3 4 5 6 7 8 9 10 Next Page

**181** Algorithm 812: BPOLY: An object-oriented library of numerical algorithms for polynomials in Bernstein form 82%
Yi-Feng Tsai , Rida T. Farouki
ACM Transactions on Mathematical Software (TOMS) June 2001 Volume 27 Issue 2

The design, implementation, and testing of a C++ software library for univriate polynomials in Bernstein form is described. By invoking the class environment and operator overloading, each polynomial in an expression is interpreted as an object compatible with the arithmetic operations and other common functions (subdivision, degree, elevation, differentiation and integration, compoistion, greatest common divisor, real-root solving, etc.) for polynomials in Bernstein form. The library allow ...

**182** Efficient on-line testing method for a floating-point adder 82%
A. Drozd , M. Lobachev
Proceedings of the DATE 2001 on Design, automation and test in Europe March 2001

**183** SP/k: a system for teaching computer programming 82%
R. C. Holt , D. B. Wortman , D. T. Barnard , J. R. Cordy
Communications of the ACM May 1977
Volume 20 Issue 5

SP/k is a compatible subset of the PL/I language that has been designed for teaching programming. The features of the SP/k language were chosen to encourage structured problem solving by computers, to make the language easy to learn and use, to eliminate confusing and redundant constructs, and to make the language easy to compile. The resulting language is suitable for introducing programming concepts used in various applications, including business data processing, scientific calculations ...

**184** FOCUS microcomputer number system 82%
Albert D. Edgar , Samuel C. Lee
Communications of the ACM March 1979
Volume 22 Issue 3

FOCUS is a number system and supporting computational algorithms especially useful for microcomputer control and other signal processing applications. FOCUS has the wide-ranging character of floating-point numbers with a uniformity of state distributions that give FOCUS better than a twofold accuracy advantage over an equal word length floating-point system. FOCUS computations are typically five times faster than single precision fixed-point or integer arithmetic for a mixture of operations ...

**185** Improving the performance of speculatively parallel applications on the Hydra CMP 82%
Kunle Olukotun , Lance Hammond , Mark Willey
Proceedings of the 13th international conference on Supercomputing May 1999

**186** Efficient support for complex numbers in Java 82%
Peng Wu , Sam Midkiff , José Moreira , Manish Gupta
Proceedings of the ACM 1999 conference on Java Grande June 1999

**187** A conversation about computer science education 82%
Daniel D. McCracken , Dennis J. Frailey
ACM SIGCSE Bulletin June 1998
Volume 30 Issue 2

This discussion began as an invitation to comment on the advisability of replacing some of the study of data structure implementation in CS2. It broadened into a discussion of a number of other aspects of computer science education. Dennis Frailey writes from an industrial point of view, although he also teaches computer science on an adjunct basis and was once a full-time academic. Dan McCracken writes from an academic point of view, although he has also worked in industry.

**188** A comparison of Ada and Java as a foundation teaching language 82%
Benjamin M. Brosgol
ACM SIGAda Ada Letters September 1998
Volume XVIII Issue 5

Java has entered the software arena in unprecedented fashion, upstaging languages and technologies that are longstanding players in the industry. Almost unheard of before 1995, the language and its surrounding technology are attracting increasing attention not merely in the hardware and software communities but also among lay users and in the popular press. This phenomenon has not escaped the attention of academia, and a growing number of colleges and universities are looking at Java as a candid ...

**189** APL to Ada translator 82%
Jack G. Rudd , Eric M. Klementis
ACM SIGAPL APL Quote Quad , Proceedings of the international conference on APL: APL in transition January 1987
Volume 17 Issue 4

**190** ftd: an exact frequency to time domain conversion for reduced order RLC interconnect models 82%
Ying Liu , Lawrence T. Pileggi , Andrzej J. Strojwas
Proceedings of the 35th annual conference on Design automation conference May 1998

Recursive convolution provides an exact solution for interfacing reduced-order frequency domain representations with discrete time domain models of piecewise linear voltage waveforms. The state-space method is more efficient, but not exact, and can sometimes produce large time domain errors. This paper presents a new algorithm, ftd (frequency to time domain), for incorporating linear frequency domain macro-models into time domain simulators. ftd provides accuracy equivalent to recur ...

**191** Symbolic mathematics system evaluators (extended abstract) 82%
Richard J. Fateman
Proceedings of the 1996 international symposium on Symbolic and

algebraic computation October 1996

**192** A Fortran 90 environment for research and prototyping of enclosure algorithms for nonlinear equations and global optimization 82%
R. Baker Kearfott
ACM Transactions on Mathematical Software (TOMS) March 1995
Volume 21 Issue 1

An environment for general research into and prototyping of algorithms for reliable constrained and unconstrained global nonlinear optimization and reliable enclosure of all roots of nonlinear systems of equations, with or without inequality constraints, is being developed. This environment should be portable, easy to learn, use, and maintain, and sufficiently fast for some production work. The motivation, design principles, uses, and capabilities for this environment are outlined. The envi ...

**193** Algorithm 737; INTLIB: a portable Fortran 77 interval standard-function library 82%
R. B. Kearfott , M. Dawande , K. Du , C. Hu
ACM Transactions on Mathematical Software (TOMS) December 1994
Volume 20 Issue 4

INTLIB is meant to be a readily available, portable, exhaustively documented interval arithmetic library, written in standard Fortran 77. Its underlying philosophy is to provide a standard for interval operations to aid in efficiently transporting programs involving interval arithmetic. The model is the BLAS package, for basic linear algebra operations. The library is composed of elementary interval arithmetic routines, standard function routines for interval data and values, and utility ro ...

**194** Gun turret drive stabilization system for the M1A2 82%
Richard E. Copra , Joseph R. Vogl
Proceedings of the conference on TRI-Ada '91: today's accomplishments; tomorrow's expectations December 1991

**195** The LINC was early and small 82%
Wesley Clark
Proceedings of the ACM Conference on The history of personal workstations January 1986

The LINC represents one of the earliest attempts to put the stored program computer into the form of a general instrument for laboratory use. In a deliberate departure from the technology of Timesharing then just beginning nearly two decades of development, the LINC was designed for use by individual experimenters and thus anticipated features of the modern personal computer and personal workstation. Built at M.I.T. in 1962, its immediate forebears were the TX-0, ARC-1, and L-1 computers, i ...

**196** Software and hardware parallelism on the iWarp multi-computer 82%
Herbert G. Mayer , Brent Baxter
Proceedings of the 5th international conference on Supercomputing June 1991

**197** Real-time data acquisition at mission control 82%
John Muratore , Troy Heindel , Terri Murphy , Arthur Rasmussen , Robert McFarland
Communications of the ACM December 1990
Volume 33 Issue 12

Perhaps one of the most powerful symbols of the United States' technological prowess is the Mission Control Center (MCC) at the Lyndon B. Johnson Space Center in Houston. The rooms at Mission Control have been witness to major milestones in the history of American technology such as the first lunar landing, the rescue of Skylab, and the first launch of the Space Shuttle. When Mission Control was first activated in the early 1960s it was truly a technological marvel. This facility, however, ...

**198** Implementation of a high level language machine 82%
A. Hassitt , J. W. Lageschulte , L. E. Lyon
Communications of the ACM April 1973
Volume 16 Issue 4

Computing machines which directly execute the statements of a high level language have been proposed in the past. This report describes the actual implementation of such a machine: it is a computer whose &ldquo;machine language&rdquo; is APL. The machine is fully operational and correctly executes almost all of the APL operations on scalars, vectors, and arrays. The machine automatically allocates memory, executes statements, calls functions, converts numbers from one type to another, check ...

**199** The accumulation buffer: hardware support for high-quality rendering 80%
Paul Haeberli , Kurt Akeley

ACM SIGGRAPH Computer Graphics , Proceedings of the 17th annual conference on Computer graphics and interactive techniques
September 1990
Volume 24 Issue 4

**200** Hardware speedups in long integer multiplication 80%
M. Shand , P. Bertin , J. Vuillemin
Proceedings of the second annual ACM symposium on Parallel algorithms and architectures May 1990

**Results 181 - 200 of 200** **short listing**

Prev Page **1 2 3 4 5 6 7 8 9 10** Next Page